TOPICAL ISSUES OF PRACTICE AND SCIENCE

Abstracts of XXVI International Scientific and Practical Conference

London, Great Britain
May 18 – 21, 2021

Library of Congress Cataloging-in-Publication Data

UDC 01.1
The XXVI International Science Conference «Topical issues of practice and science», May 18 – 21, 2021, London, Great Britain. 835 p.

ISBN - 978-1-63848-661-9

DOI - 10.46299/ISG.2021.I.XXVI

EDITORIAL BOARD

FEATURES OF SOFTWARE APPLICATION DEVELOPMENT FOR FOOD RECOGNITION USING DEEP MACHINE LEARNING METHODS

Tvoroshenko Iryna,
Ph.D., Associate Professor
Kharkiv National University of Radio Electronics

Temchur Karyna,
Bachelor in informatics
Kharkiv National University of Radio Electronics

One of the most interesting modern problems that researchers of computer vision are trying to solve is the recognition and classification of objects in images [1-6].

An interesting example in this area is recognition in images of food and culinary dishes.

With the modern pace of life, with the spread of a sedentary lifestyle, a variety of foods with different levels of useful substances, constant haste, digestive diseases, and the problem of excess body fat, many people are increasingly thinking about how to control, what are they consume. But most of the systems used to control nutrition are based on an outdated approach when all the data must be searched and entered manually [7-10].

The work aims to develop a software application using the methods of deep machine learning, aimed at recognifying and classifying culinary dishes.

However, most researchers prefer to use convolutional neural networks or CNN. CNN has successfully used in its research the recognition and classification of culinary dishes by Japanese researchers [11], Milan researchers [12], Singaporean researchers who developed the software application FoodAI [13], and even researchers from Google developing the complex system Im2Calories [14].

Convolutional neural network or CNN refers to a type of multilayer network of direct propagation, in which the signal propagates in one direction. The principle of operation of the network is inspired by the work of the visual cortex of animals and aims to effectively image recognition [15].

CNN is part of deep learning technologies and is part of many deep learning libraries, such as TensorFlow.

The idea of the CNN architecture is to alternate convolutional and sub discrete (subsample) layers. The network got its name from the convolution operation, in which each fragment of the image is multiplied by the convolution core element by element [16].

TensorFlow is an open-source machine and deep learning library developed and introduced by Google in 2015 [17]. It was designed specifically to solve the problems of pattern recognition and classification with the highest level of efficiency.

Therefore, to develop an application for the recognition and classification of culinary dishes, it was decided to use the following technologies:

– TensorFlow, TensorFlow.js, Keras;
– Python;
– TypeScript, React, Express.js, Redux, Webpack;
– Dataset Food101;
– The open model is Inceptionv3.

The first step in developing the application was to work with data sampling.

A ready sample of Food101 with 101 classes of dishes was taken as data. In total, the set contains 101000 images of culinary dishes.

An example of images from the training set is shown in Figure 1.

Figure 1. Example of images from the training set.

After unpacking all the files, you need to divide the images into training and test. You can then proceed to adjust the weights of the pre-trained Inceptionv3 model using the Food101 sample.

As it was stated, the use of a pre-trained model with its already determined weights will speed up the learning process and save not only time but also computing resources. For further work, you only need to adjust the scales of the model to the sample and add a few new layers, instead of creating everything from scratch.

We create a new function train_model, in which we create a session. After that, we carry out further processing of images: we reduce all images to the size of 300×300. For training and test images, we generate different variations: images with zoom, rotation, and shift.

Next, initialize the Inceptionv3 model. We set for it the ReLu activation function, which was mentioned in previous sections.

At the same time, the cross-entropy loss function Softmax, which was also described earlier, is taken as a loss function. Stochastic gradient descent or SGD is taken as the neural network optimizer.

After tweaking the model, sets the number of classes.

In this case, we have a 101 class. We establish epochs, or in other words – iterations of training.

In such a way, the resource-intensive process of learning the model begins. Each epoch, data from the training sample pass through the model, and the accuracy of training is analyzed based on test data. Each iteration increases the accuracy of model recognition and classification.

The learning outcomes are shown in Figure 2.

```
Epoch 1/20
4734/4734 [==============================] - 17810s 4s/step - loss: 5.0469 - accuracy: 0.0412 - val_loss: 3.7701 - val_accuracy: 0.3359

Epoch 00001: val_loss improved from inf to 3.77012, saving model to bestmodel_101class.hdf5
Epoch 2/20
4734/4734 [==============================] - 17791s 4s/step - loss: 3.8104 - accuracy: 0.2786 - val_loss: 2.4422 - val_accuracy: 0.5539

Epoch 00002: val_loss improved from 3.77012 to 2.44223, saving model to bestmodel_101class.hdf5
Epoch 3/20
4734/4734 [==============================] - 17808s 4s/step - loss: 2.9014 - accuracy: 0.4360 - val_loss: 1.9032 - val_accuracy: 0.6463

Epoch 00003: val_loss improved from 2.44223 to 1.90316, saving model to bestmodel_101class.hdf5
Epoch 4/20
4734/4734 [==============================] - 17818s 4s/step - loss: 2.4352 - accuracy: 0.5235 - val_loss: 1.6446 - val_accuracy: 0.6968

Epoch 00004: val_loss improved from 1.90316 to 1.64460, saving model to bestmodel_101class.hdf5
Epoch 5/20
4734/4734 [==============================] - 17821s 4s/step - loss: 2.1539 - accuracy: 0.5820 - val_loss: 1.4584 - val_accuracy: 0.7343

Epoch 00005: val_loss improved from 1.64460 to 1.45843, saving model to bestmodel_101class.hdf5
Epoch 6/20
4734/4734 [==============================] - 17813s 4s/step - loss: 1.9443 - accuracy: 0.6229 - val_loss: 1.3502 - val_accuracy: 0.7540

Epoch 00006: val_loss improved from 1.45843 to 1.35024, saving model to bestmodel_101class.hdf5
Epoch 7/20
4734/4734 [==============================] - 17812s 4s/step - loss: 1.7936 - accuracy: 0.6541 - val_loss: 1.2665 - val_accuracy: 0.7685

Epoch 00007: val_loss improved from 1.35024 to 1.26646, saving model to bestmodel_101class.hdf5
Epoch 8/20
4153/4734 [=========================>....] - ETA: 34:53 - loss: 1.6643 - accuracy: 0.6792
```

Figure 2. Learning outcomes of the model.

As you can see, the model has passed less than 8 epochs out of 20 installed and stopped. The learning process for less than 8 epochs took almost 5 hours.

At a certain point, almost at the end of the 8th epoch, model learning came to a halt due to a lack of computing resources.

However, 8 incomplete epochs were enough to get 77% accuracy of the model, which is a pretty good result.

After the training process, the client and server part of the application is developed using TypeScript, React, Express.js, Redux, Webpack, and TensorFlow.js technologies to convert the model from Keras to JSON format.

The final type of application is shown in Figure 3.

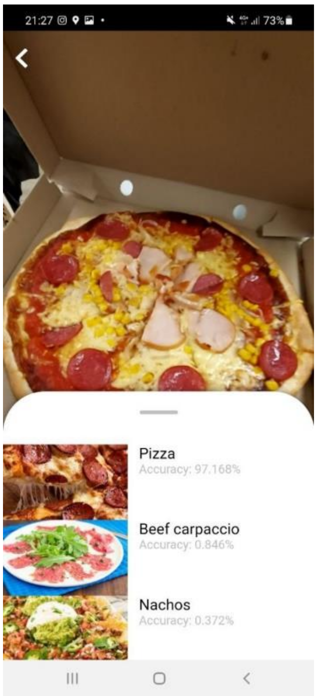

Figure 3. Screen form of the finished application.

The finished application has been tested on various data. The result with 77% accuracy is quite good but indicates a fairly high percentage of errors in recognition and classification. This means that there are cases when the model does not correctly recognize the object in the image. For example, depending on the color, image quality, and contrast, the application may define a “steak” as a “chocolate cake” and vice versa.

There are currently several promising ideas for further application improvement:

– Create subclasses for existing general classes of dishes. That is, instead of classifying a dish by general class, delve into finding its subclass. For example, if a dish belongs to the general class “pizza”, then determine its subclass in this class. For example, “pizza with seafood”;

– Add new classes to be able to recognize more dishes;

– Add information about CPFC (Calorie, Protein, Fats, Carbohydrate) to food classes;

– Increase the accuracy of recognition.

References:

1. Gorokhovatskyi V., and Tvoroshenko I. (2020) Image Classification Based on the Kohonen Network and the Data Space Modification, *In CEUR Workshop Proceedings: Computer Modeling and Intelligent Systems*, 2608, pp. 1013-1026.

2. Gorokhovatskyi V.O., Tvoroshenko I.S., and Vlasenko N.V. (2020) Using fuzzy clustering in structural methods of image classification, *Telecommunications and Radio Engineering*, 79(9), pp. 781-791.

3. Kobylin O., Gorokhovatskyi V., Tvoroshenko I., and Peredrii O. (2020) The application of non-parametric statistics methods in image classifiers based on structural description components, *Telecommunications and Radio Engineering*, 79(10), pp. 855-863.

4. Daradkeh, Y.I., Tvoroshenko, I., Gorokhovatskyi, V., Latiff, L.A., and Ahmad, N. (2021) Development of Effective Methods for Structural Image Recognition Using the Principles of Data Granulation and Apparatus of Fuzzy Logic, *IEEE Access*, 9, pp. 13417-13428.

5. Gorokhovatskyi V.O., Tvoroshenko I.S., and Peredrii O.O. (2020) Image classification method modification based on model of logic processing of bit description weights vector, *Telecommunications and Radio Engineering*, 79(1), pp. 59-69.

6. Gorokhovatskyi, V., Rusakova, N., and Tvoroshenko, I. (2020) The application of image analysis methods and predicate logic in applied problems of magnetic monitoring, *Telecommunications and Radio Engineering*, 79(20), pp. 1801-1811.

7. M. Ayaz Ahmad, Irina Tvoroshenko, Jalal Hasan Baker, Liubov Kochura, Vyacheslav Lyashenko (2020) Interactive Geoinformation Three-Dimensional Model of a Landscape Park Using Geoinformatics Tools, *International Journal on Advanced Science, Engineering and Information Technology*, 10(5), pp. 2005-2013.

8. Nong Ye. (2013) *Data Mining: Theories, Algorithms, and Examples*, Florida, USA: CRC Press, 349 p.

9. Tvoroshenko I.S., and Gorokhovatsky V.O. (2019) Modification of the branch and bound method to determine the extremes of membership functions in fuzzy intelligent systems, *Telecommunications and Radio Engineering*, 78(20), pp. 1857-1868.

10. Tvoroshenko I.S., and Gorokhovatsky V.O. (2020) Effective tuning of membership function parameters in fuzzy systems based on multi-valued interval logic, *Telecommunications and Radio Engineering*, 79(2), pp. 149-163.

11. Kagaya H., Aizawa K., and Ogawa M. (2014, November) Food detection and recognition using convolutional neural network. *In Proceedings of the 22nd ACM international conference on Multimedia*, pp. 1085-1088.

12. Ciocca G., Napoletano P., and Schettini R. (2016) Food recognition: a new dataset, experiments, and results. *IEEE journal of biomedical and health informatics*, 21(3), pp. 588-598.

13. Sahoo D., Hao W., Ke S., Xiongwei W., and et al. (2019, July) FoodAI: Food image recognition via deep learning for smart food logging. *In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2260-2268.

14. Meyers A., Johnston N., Rathod V., Korattikara A., and et al. (2015) Im2Calories: towards an automated mobile vision food diary. *In Proceedings of the IEEE International Conference on Computer Vision*, pp. 1233-1241.

15. Liu Y. H. (2018, September) Feature extraction and image recognition with convolutional neural networks. *In Journal of Physics: Conference Series*, 1087(6), p. 062032.

16. Duda R.O., Hart P.E., and Stork D.G. (2000) *Pattern classification*, Hoboken, USA: John Wiley & Sons, 738 p.

17. Peters J.F. (2017) *Foundations of computer vision: Computational Geometry, Visual Image Structures and Object Shape Detection*, Cham, Switzerland: Springer International Publisher, 417 p.